

Problem Statement:

A requirement from the Hospital, Management asked us to create a predictive model which will predict the Chronic Kidney Disease (CKD) based on the several parameters. The Client has provided the dataset of the same.

Identification of problem statement:

All the input data are well defined and out to be predict is clearly defined so we can proceed with supervised learning (ML) with using classification algorithm since the predictable data is categorical data.

Basic info about the dataset:

The dataset contains 400 rows and 25 columns. Some columns contain categorical data, so we have to convert it to numerical using one hot encoding.

pre-processing method:

The input data are pre-processed by Standard-Scaler algorithm and label-Encoder if necessary.

Evaluation metric for all possible models:

SVC:

	precision	recall	f1-score	support
0	0.98	1.00	0.99	51
1	1.00	0.99	0.99	82
accuracy			0.99	133
macro avg	0.99	0.99	0.99	133
weighted avg	0.99	0.99	0.99	133

```
f1=f1_score(y_test,y_pred,average='weighted')
f1
```

```
0.9924946382275899
```

```
roc_auc_score(y_test,grid.predict_proba(x_test)[:,1])
```

```
1.0
```

Decision Tree Classifier:

	precision	recall	f1-score	support
0	0.96	1.00	0.98	51
1	1.00	0.98	0.99	82
accuracy			0.98	133
macro avg	0.98	0.99	0.98	133
weighted avg	0.99	0.98	0.99	133

```
f1=f1_score(y_test,y_pred,average='weighted')  
f1
```

0.9850141736106648

```
roc=roc_auc_score(y_test,grid.predict_proba(x_test)[:,1])  
roc
```

0.9878048780487805

Random Forest Classifier:

	precision	recall	f1-score	support
0	0.00	0.00	0.00	51
1	0.62	1.00	0.76	82
accuracy			0.62	133
macro avg	0.31	0.50	0.38	133
weighted avg	0.38	0.62	0.47	133

C:\Users\shanm\Anaconda3\lib\site-packages\sklearn\metrics\classification_report.py:130: UserWarning: Some of the classes are ill-defined and being set to 0.0 in labels with no predicted samples.
'precision', 'predicted', average, warn_for)

```
f1=f1_score(y_test,y_pred,average='weighted')  
f1
```

C:\Users\shanm\Anaconda3\lib\site-packages\sklearn\metrics\classification_report.py:130: UserWarning: Some of the classes are ill-defined and being set to 0.0 in labels with no predicted samples.
'precision', 'predicted', average, warn_for)

0.47029200909249863

```
roc=roc_auc_score(y_test,grid.predict_proba(x_test)[:,1])  
roc
```

0.96987087517934

Logistic_Regression:

	precision	recall	f1-score	support
0	0.98	1.00	0.99	51
1	1.00	0.99	0.99	82
accuracy			0.99	133
macro avg	0.99	0.99	0.99	133
weighted avg	0.99	0.99	0.99	133

```
f1=f1_score(y_test,y_pred,average='weighted')  
f1
```

0.9924946382275899

```
roc_auc_score(y_test,grid.predict_proba(x_test)[: ,1])
```

1.0

KNN:

	precision	recall	f1-score	support
0	0.91	1.00	0.95	51
1	1.00	0.94	0.97	82
accuracy			0.96	133
macro avg	0.96	0.97	0.96	133
weighted avg	0.97	0.96	0.96	133

```
f1=f1_score(y_test,y_pred,average='weighted')  
f1
```

0.9626932787797391

```
roc_auc_score(y_test,grid.predict_proba(x_test)[: ,1])
```

0.998804399808704

By considering the F1_score, accuracy, roc_auc_score and recall and precision of individual class we can select SVC or Logistic Regression as a best Model.